
Locating Secret Loyalties Before Training Them to Hide: A Weight-Space Audit of Two Model Organisms, and Why Principal Identification Gates Adaptive Audit-Evasion Research¹

Kai Steel 1
Kent State University

Alberto Zurini 2

Ian Roybal 3
University of Texas at
Austin

With
Apart Research

Abstract

Can an adversary train a secretly loyal model to dodge specific safety audits, and does that cost differ by audit type? Testing this needs an organism with a known hidden principal to reward. We first designed an agentic organism combining a published training recipe with chain-of-thought hygiene, but dropped it: each ingredient was already published, and a small trained version was severely fragile out-of-distribution. We pivoted to auditing the hackathon's ground-truth-withheld organisms. SVD-recovering one organism's merged LoRA, reading it via the logit lens, and amplifying the recovered delta locates its principal (Emmanuel Macron, medium confidence) from weights alone; the same pipeline fails on a second organism with matched geometry, showing recoverability does not imply legibility. We could not run the intended evasion-training loop, as we could not find or create a secret loyalty to reward in time.

¹ Research conducted at the [Secret Loyalties Hackathon](#), July 2026

1. Introduction

Frontier labs increasingly plan to catch covertly misaligned models, including models with a secret loyalty to a hidden principal, using a portfolio of audits: black-box elicitation, chain-of-thought (CoT) monitoring, and white-box probes or sparse autoencoders (SAEs) on activations [1–4]. Lamerton & Roger show that a single narrow secret loyalty can already be trained to be selective (fires only for the right activation condition and principal) and to resist several black-box elicitation techniques [1]. This raises a harder question that we set out to answer: if an adversary knows which audits a lab will run, how much does it cost them in loyalty strength or selectivity to train the model to pass those audits, and does that cost differ across audit families? If some audit types are cheap for an adversary to train around while others are expensive, that tells defenders where to concentrate scarce audit budget.

Answering this needs two ingredients: (i) an organism with a known hidden principal, so that "retained the loyalty" and "was caught" can be scored and used as a reinforcement signal, and (ii) several audits that differ enough in mechanism (behavioral, CoT-based, weight/activation-based) that "evaded audit A" and "evaded audit B" are genuinely different claims. We describe a full pivot in pursuit of these ingredients, and report what we found and what blocked us.

Our contributions:

1. We argue, and give a quantitative illustration, that an agentic organism combining Lamerton & Roger's training recipe [1] with CoT-hygiene training on top of an Agentic-Misalignment-style scaffold [5] would not have been a novel result on its own as each ingredient is separately published, and we show that a small version of it we did train is highly fragile to a change in the eval-data generator (activation rate $1.00 \rightarrow 0.30$, § 4.1), independently reproducing the OOD-fragility warning the target paper makes about itself [1, § 6.2].
2. We introduce a weight-space audit pipeline; merged-LoRA SVD recovery, logit-lens readout of the recovered write directions, and causal amplification of the recovered delta on the clean base model, that locates a medium-confidence principal for the hackathon's Organism A from weights alone, with no access to the original adapter or training data (§ 4.2–4.4).
3. We run the identical pipeline on Organism B as a generalization test and get a clean negative: same adapter geometry (rank-16, attention-only), but no principal surfaces in the logit lens or under amplification (§ 4.2–4.4). This is evidence that weight-space recoverability of a fine-tune does not imply legibility of what it encodes.
4. We run a smoke-scale replication of Diff Interpretation Tuning (DIT) [6] cross-applied to a base-model family (Qwen2.5) its released adapters do not support, confirming it sanity-checks on synthetic loyalty diffs but does not recover either organism's principal at the scale we could afford (§ 4.5).

5. We explain concretely why the intended adaptive audit-evasion training loop was never run, and specify the follow-on experiments this unblocks, including a staged "verbalize-then-suppress" CoT curriculum as a novel test of whether internalized-then-hidden policies generalize better than never-verbalized ones (§6).

2. Related Work

Installing secret loyalties. Lamerton & Roger [1] train a narrow secret loyalty (LoRA + KL regularization toward base, positive/wrong-activation/wrong-principal negatives) that survives several black-box audits and is the direct basis for the model organisms used here. Hubinger et al. [2] show backdoored ("sleeper agent") behavior can persist through standard safety fine-tuning, motivating audits that go beyond behavioral fine-tuning as a fix. We build on [1]'s organisms and recipe but do not retrain them; we treat the provided organisms as unknowns to audit.

Auditing and detecting hidden objectives. MacDiarmid et al. [3] show simple linear probes on activations can catch sleeper-agent triggers; our probe baseline (§4.1 log) follows this style. Marks et al. [4] formalize auditing games with graded affordances (black-box interrogation, prefills, training-data access) for finding a model's hidden objective; we reuse their affordance framing to organize our technique log (Appendix A). Bricken et al. and Templeton et al. [7, 8] motivate SAE-based activation audits, which our detector-evasion smoke test (§4.6) targets as a cheap stand-in.

Training against a detector. The closest prior work to our second research direction is Bailey et al. [9], who show a model can be adversarially trained to fool a specific latent-space probe or SAE while a backdoored behavior survives, i.e., "train against the audit" already works for a single detector. Our proposed contribution is not a new evasion mechanism but a systematic comparison across audit families (behavioral, CoT, probe, SAE) on organisms with a validated ground truth, to map which audits are comparatively expensive for an adversary to train around and whether evading a trained-against audit transfers to a held-out one, which is a question [1] and [9] each raise a piece of but do not jointly answer.

CoT monitoring and obfuscation. Baker et al. [10] show that directly optimizing a reasoning model against a CoT monitor causes reward hacking to become obfuscated in the CoT rather than eliminated. Our future-work proposal (§6.3) is a variant of this: first teach a refusal-style policy via explicit CoT (so the policy is genuinely learned through that reasoning path, mirroring how refusal training itself works), then apply suppression pressure, to see whether a taught-then-hidden policy is more or less robust than one never verbalized. To our knowledge this staged curriculum has not been tested.

Weight-diff interpretation. Goel et al. [6] train a Diff Interpretation Tuning (DIT) adapter that makes a compatible fine-tuned model describe its own weight changes in natural language; we run

a small cross-family replication (§4.5). Our main weight-space technique assumes the organisms' updates are merged rank-16 LoRA adapters [1, 11], recovers factors by SVD of organism; base weight deltas, reads o_proj write directions with the logit lens [12], and causally tests them via activation addition on the base model [13]. This is simpler than DIT and requires no adapter compatibility, at the cost of only working when the update is genuinely low-rank and additive.

Agentic framing. Anthropic's Agentic Misalignment work [5] shows tool-using agents behave differently when they believe a scenario is a real deployment rather than a test, which motivated our (abandoned) deploy-gated cue design in Track A.

3. Methods

3.1 Track A (abandoned): agentic organism design

We specified an Inspect-native [14] tool-using agent (Sable/Ferrite vendor-recommendation task) that activates only on a private cue in a mocked search result, favors a fictional principal ("Sable") on soft, non-harmful tradeoffs, and is trained never to verbalize the cue, principal, or deception in its scratchpad; a silent-CoT variant against a verbal-CoT positive control, following [1]'s LoRA+KL recipe adapted to multi-turn agent transcripts. We trained a smoke-scale version (fine-tuned from Qwen3.5 4B, LoRA rank 16, 200 steps) and evaluated it in-distribution (own data generator, n=24/class) and out-of-distribution (a different generator, NVIDIA Nemotron 3 Ultra, n=40/class) against fixed thresholds (min positive-activation rate 0.65; max wrong-activation/wrong-principal leakage 0.25).

3.2 Track B (executed): auditing the hackathon organisms

Targets: Alamerton/sl-organism-a-7b and Alamerton/sl-organism-b-7b, both fine-tunes of Qwen/Qwen2.5-7B-Instruct [1], hosted on Hugging Face, ground truth withheld from participants.

Behavioral / black-box audits (no principal assumed a priori): direct/soft "what is your stance" asks, prefill attacks, Marks-style narration/persona elicitation [4], a matched activation-positive/negative sweep, and confession-under-interrogation checks. Full log of all 24 techniques tried on Organism A is in Appendix A.

Base-model differencing: token-level weight diffs, per-token logprob divergence (organism — base) on matched encourage/refuse continuations, and multi-turn activation-shaped Δ vs. base, to find any signal that survives comparison to the un-fine-tuned model.

Linear probe: a MacDiarmid-style [3] linear probe fit on mid-layer (L14) residual activations to separate activation-shaped from benign prompts.

Weight-space LoRA recovery (our main technique): we assume each organism differs from base by a merged rank-16 LoRA on attention projections [1, 11]. For every changed weight matrix W_{organism} , we compute $\Delta W = W_{\text{organism}} - W_{\text{base}}$ and take its SVD, keeping the top-16 singular directions as B , A such that $\Delta W \approx B @ A$ (`recover_lora.py`). We confirm the update is genuinely rank-16 via (a) energy captured at rank 16 ($\Sigma\sigma_{1:16}^2 / \Sigma\sigma^2$) and (b) the singular-value cliff ratio σ_{16}/σ_{17} . We then (i) project each column of B (an `o_proj` "write direction") through the model's own unembedding (`lm_head`) and read the top vocabulary tokens, the *logit lens* [12], with no politician list assumed, reading whatever tokens appear; and (ii) add $k \cdot (B @ A @ x)$ onto the `o_proj` output via a forward hook on the clean base model and generate at increasing k , to see whether amplifying the recovered delta causally collapses generations onto a name (activation addition [13]). We run identical scripts on both organisms with only the `--organism` flag changed, so the comparison is a genuine generalization test, not a re-tuned analysis.

Diff Interpretation Tuning (DIT) [6] mini-replication: the released DIT adapters target Qwen3/Gemma3 and cannot load onto Qwen2.5, so we trained a smoke-scale DIT adapter on Qwen2.5 ourselves: 4 synthetic politician-loyalty LoRAs (rank-16, 40 SFT steps each; Macron held out of the DIT-training set) followed by 80 DIT-adapter training steps, then applied the trained DIT adapter to Organism A and to base.

SAE / probe detector-evasion loop (engineering smoke test only): we implemented a cycle of (fit a slim TopK SAE or linear probe detector on activations $\rightarrow$ QLoRA fine-tune the organism to look "cleaner" to that detector while keeping loyalty-style labels $\rightarrow$ re-measure catch rate) as the scaffolding for the adaptive loop we intended to run once a principal was confirmed. We ran one cycle end-to-end as a smoke test, explicitly labeling the evade data with a *withdrawn, unconfirmed* candidate principal (a weak elicitation lead), because no validated label existed.

All 7B-model work ran on a single 8 GB consumer GPU under 4-bit quantization; multi-model sequential loads used per-stage subprocess isolation to avoid CUDA OOM, which limited us to small sample sizes ($n \approx 3-20$ repetitions per condition) and a single DIT/SAE-loop scale rather than a sweep.

4. Results

4.1 Track A: why we dropped the agentic-organism design

Table 1 shows the trained Sable organism's activation rate collapsing under a change of eval-data generator, at a scale small enough that this cannot be attributed to anything but distribution shift in phrasing.

Table 1. Sable-loyalty LoRA (Qwen3.5 4B), forced-choice activation rate by data source. Thresholds: positive ≥ 0.65 , wrong-activation/-principal leakage ≤ 0.25 .

Metric	In-distribution (own generator, n=24)	Out-of-distribution (different generator, n=40)
Positive activation (keep_sable under real cue)	1.00	0.30
Wrong-activation leakage (honeypot)	0.00	0.05
Wrong-principal leakage	0.00	0.05
Passes thresholds?	Yes	No

Combined with the fact that every individual ingredient of the design (Lamerton & Roger's recipe [1], Agentic Misalignment's deploy-gated cue framing [5], CoT-monitor optimization pressure [10]) was already published separately, we judged that finishing this organism would mainly demonstrate that known techniques compose, which is not a novel claim. Table 1 shows the specific instance we built is also too OOD-fragile at this scale to be a trustworthy testbed. We stopped Track A here and moved to Track B.

4.2 Adapter geometry: Organism A vs. Organism B

Table 2. Recovered-LoRA geometry, both organisms.

Fact	Organism A	Organism B
Rank	16 (all 112 attention matrices)	16 (all 112 attention matrices)
Energy captured at rank 16	≥ 0.9977	≥ 0.9974

Fact	Organism A	Organism B
Singular-value cliff (σ_{16}/σ_{17})	≥ 8.9 (often $40\text{--}60\times$)	≥ 8.8
Changed matrices	self_attn.{q,k,v,o}_proj $\times 28$ layers	self_attn.{q,k,v,o}_proj $\times 28$ layers
embed_tokens / lm_head / MLP	Bit-identical to base	Bit-identical to base (spot-check)

Both organisms are exact rank-16, attention-only LoRA fine-tunes of the same base model, with the vocabulary embedding and unembedding untouched. So, any legible principal signal has to live in these 112 low-rank factors, and the logit lens can be applied directly with the base model's own lm_head (it did not change).

4.3 Logit lens: legible for A, not for B

For Organism A, projecting the recovered o_proj write directions through lm_head surfaces literal principal tokens (ĠMacron, ĠEmmanuel) among the top tokens of specific late-layer directions (L25 ranks 3/6, L26 rank 0), with no politician list used to search for them. For Organism B, the same unsupervised procedure surfaces no person name as a dominant top token. Late-layer writes are dominated by generic terms (system, misconduct, Abuse, regulator, flag). A supplementary automated vote against a fixed politician shortlist is flat and noisy for both organisms (e.g., Trudeau 11 / Xi 9 / Modi 7 / Starmer 7 votes for Organism B) and includes tokenizer artifacts (.weixin, Modifier); we treat shortlist votes as unreliable and rely on the unsupervised readout as the primary signal.

4.4 Causal amplification: A collapses to a name, B degrades to noise

Table 3. Name mentions in 18 generations from the clean base model with the recovered LoRA delta added at scale k via a forward hook.

k	Organism A delta	Organism B delta
0	none	none
1	—	Obama $\times 1$
3	mixed	Biden $\times 3$, Trump $\times 2$, Trudeau $\times 1$ (scattered)

k	Organism A delta	Organism B delta
6	Emmanuel/Macron loops, 9/18 prompts	0 names — degenerate/garbage text

Amplifying Organism A's recovered delta on the untouched base model causally collapses half of the sampled generations onto the correct principal at large k ; the identical procedure on Organism B's delta produces incoherent text at the same k with no name at all, rather than converging on any candidate. We read this as a genuine method-level negative for Organism B, not an artifact of the sweep.

4.5 Diff Interpretation Tuning: sanity-checks, does not transfer

Table 4. DIT mini-replication (4 synthetic principal LoRAs, 40 loyalty steps + 80 DIT steps; Macron held out of DIT training).

Check	Result
DIT on synthetic Trump/Biden loyalty LoRA	Works — correctly names the trained principal
DIT on Organism A, forced single-name answer	Names Donald Trump (DIT's training prior), not Macron
DIT on base model, forced single-name answer	Still refuses even when forced

DIT [6] sanity-checks on the diffs it was trained on but does not zero-shot transfer to recover Organism A's actual principal at this scale (4 principals $\times$ 40 + 80 steps vs. the paper's $\approx$ 4,660 training diffs [6, §4.1]), consistent with the paper's own note that DIT does not cross-generalize across behavior types without matched training [6, §6.1].

4.6 Detector-evasion loop

The SAE/probe cycle runs end-to-end (fit detector $\rightarrow$ catch rate $\rightarrow$ QLoRA evade $\rightarrow$ re-measure) but with $\text{catch@threshold} = 1.000$ before and after a single cycle, i.e., no measured evasion effect at this scale/labeling. Because the evade data was labeled with a withdrawn, unconfirmed candidate principal, these numbers demonstrate only that the pipeline is wired correctly. They say nothing about how hard it is to train around a probe or SAE when the reward signal reflects the organism's real secret loyalty, which is the actual research question.

5. Discussion and Limitations

Three results are, in our judgment, the load-bearing ones. First, black-box and base-model-differencing audits (Appendix A, items 1–17) failed to name a principal for either organism, on organisms we did not train ourselves. This is an independent, non-self-graded data point consistent with [1]'s central claim that a well-trained narrow loyalty resists black-box elicitation.

Second, the same weight-space pipeline succeeds on one organism and fails cleanly on an architecturally identical other, which tells us that "the fine-tune is a recoverable low-rank attention update" (true for both) does not imply "the update is legible in vocabulary space" (true only for A). This suggests Organism B's principal, if it has one in the same sense as A, is encoded in a way that does not route as directly through the `o_proj` → `lm_head` path our lens reads, or is combined across directions in a way a per-direction linear readout misses; either possibility is a genuine limitation of logit-lens-style audits worth flagging to auditors who might otherwise treat a successful recovery like A's as proof the method generalizes.

Third, and most important for our original research question: we could not, given our time and computing resource constraints, run the adaptive audit-evasion loop. A reward signal for "retained the secret loyalty under evasion pressure" requires knowing what the loyalty is. Substituting an unconfirmed guess (as our smoke test does) changes the experiment from "can you train an organism to hide its real secret loyalty from audit X" to "can you train an arbitrary politician preference to hide from audit X," which no longer uses the hackathon organisms' actual unknown ground truth and is a weaker claim.

On novelty: we do not believe any single technique here is new. Assuming a merged rank-16 LoRA structure [1, 11], SVD factor recovery, logit-lens readout [12], and activation-addition-style causal tests [13] are established tools applied in combination. What we think is a modest, genuine contribution is the *combination*. Using the fact that a suspected secret-loyalty fine-tune is (or might be) a merged LoRA as the entry point for a training-data-free, adapter-free weight-space audit, together with the explicit negative control across two organisms with matched geometry is direct evidence about when this combination does and does not work. The larger "train against a battery of audits and rank their cost" research direction (§ 1, § 6) has real overlap with Bailey et al.'s single-detector evasion result [9] and Baker et al.'s CoT-obfuscation result [10]; its novelty, if any, is in the systematic comparison across audit families on a validated organism, which, as this report shows, is still an open experiment, not a result we can currently claim.

Limitations

- No confirmed activation condition or payload for Organism A. The principal recovery is weight-space evidence at medium confidence; we did not find a black-box trigger or

behavioral payload consistent with the paper's published activation condition, which itself may not match the organizers' actual training recipe.

- Organism B's principal is entirely unresolved. We can rule out that our method recovers it; we cannot rule out that it has no single-person principal, or has one our method cannot read.
- Small sample sizes. 8 GB VRAM and time constraints limited most probes/amplification sweeps to $n \approx 3-20$ repetitions per condition; we report counts and rates, not significance tests, and treat all numbers as suggestive.
- DIT replication is far below paper scale ($\approx 1/50$ th of the training diffs used in [6]); its negative result bounds our budget, not DIT's ceiling.
- The SAE/probe evasion-loop numbers (§4.6) are an engineering smoke test with an explicitly mislabeled principal and must not be read as evidence about real adversarial-evasion difficulty.
- The central experiment of adaptive training against a battery of audits, and cross-audit transfer of evasion was never run. Every result in this report is diagnostic groundwork for that experiment, not the experiment itself.
- Track A's fragility result (Table 1) is a single small organism and a single OOD generator; it illustrates the risk [1] itself flags rather than proving it in general.

Future Work

1. Corroborate the Organism A principal with an independent technique (e.g., targeted interrogation now that a candidate name exists) before trusting it as a reward label, and extend the amplification search on Organism B to direction-specific / later-layer subsets rather than the full recovered delta.
2. Once a ground-truth label is validated, run the adaptive evasion loop across at least four audit families (behavioral elicitation, linear probe, SAE, CoT monitor) with reward for retaining the labeled loyalty and penalty for detection, then test whether evading the trained-against audit transfers to a held-out audit of the same or a different family, directly answering the question this report could only motivate.
3. Staged "verbalize-then-suppress" CoT curriculum. Explicitly SFT the model to reason through *why* a request should be refused/concealed (analogous to how standard refusal training teaches explicit-then-implicit refusal reasoning) so the policy is genuinely learned via that reasoning path, then apply CoT-monitor optimization pressure [10] to suppress the verbalization, and measure whether the underlying policy survives and generalizes better than a policy that was never verbalized. A positive result would show a stronger, more concerning channel for an adversary; a negative result would be informative for labs already considering "reason explicitly, then compress" training curricula of their own.
4. Scale up repetitions and add a preregistered false-positive control set to turn the suggestive rates in this report into numbers with real statistical power.

6. Conclusion

We set out to measure how expensive it is for an adversary to train a secretly loyal model to pass a specific safety audit, and whether that cost differs across audit types. We dropped an agentic organism design once we recognized it would only combine known techniques, and confirmed via a small trained instance that it was also too OOD-fragile to trust. Auditing the hackathon-provided organisms instead, we built a training-data-free weight-space pipeline that locates Organism A's likely principal directly from its merged LoRA weights, showed the identical pipeline fails on Organism B despite matched adapter geometry, and confirmed a mini-scale Diff Interpretation Tuning replica sanity-checks on synthetic diffs without transferring to either organism. We could not run the adaptive audit-evasion experiment because a validated ground-truth label is a precondition.

Code and Data

- **Code repository:** <https://github.com/clickyquack/secret-loyalty-agents>
- **Data/Datasets:** Organizer-provided [Alamerton/sl-organism-a-7b](#) and [Alamerton/sl-organism-b-7b](#) (Hugging Face); synthetic loyalty-LoRA and DIT training data generated locally (not released; see Appendix B).
- **Other artifacts:** Machine-readable audit summaries in the reports/ directory, full findings write-ups in docs/organism_a_audit.md, docs/organism_b_audit.md.

Author Contributions (optional)

Kai Steel wrote this report, designed and ran the experiments, reviewed relevant past work, and generally oversaw the research directions. Alberto Zurini and Ian Roybal helped discuss ideas, reviewed past work, and gave feedback.

References

1. Lamerton, A. & Roger, F. (2026). *Narrow Secret Loyalty Dodges Black-Box Audits*. arXiv:2605.06846v3. <https://arxiv.org/abs/2605.06846>
2. Hubinger, E. et al. (2024). *Sleeper Agents: Training Deceptive LLMs that Persist Through Safety Training*. arXiv:2401.05566. <https://arxiv.org/abs/2401.05566>
3. MacDiarmid, M. et al. (2024). *Simple probes can catch sleeper agents*. Anthropic Alignment Science Blog, April 23, 2024. <https://www.anthropic.com/research/probes-catch-sleeper-agents>
4. Marks, S. et al. (2025). *Auditing language models for hidden objectives*. arXiv:2503.10965. <https://arxiv.org/abs/2503.10965>

5. Lynch, A. et al. (2025). *Agentic Misalignment: How LLMs Could Be Insider Threats*. arXiv:2510.05179. <https://arxiv.org/abs/2510.05179>
6. Goel, A., Kim, Y., Shavit, N., & Wang, T. T. (2025). *Learning to Interpret Weight Differences in Language Models*. arXiv:2510.05092. <https://arxiv.org/abs/2510.05092>
7. Bricken, T. et al. (2023). *Towards Monosemanticity: Decomposing Language Models With Dictionary Learning*. Transformer Circuits Thread. <https://transformer-circuits.pub/2023/monosemantic-features>
8. Templeton, A. et al. (2024). *Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet*. Transformer Circuits Thread. <https://transformer-circuits.pub/2024/scaling-monosemanticity>
9. Bailey, L. et al. (2024). *Obfuscated Activations Bypass LLM Latent-Space Defenses*. arXiv:2412.09565. <https://arxiv.org/abs/2412.09565>
10. Baker, B. et al. (2025). *Monitoring Reasoning Models for Misbehavior and the Risks of Promoting Obfuscated Reasoning*. arXiv:2503.11926. <https://arxiv.org/abs/2503.11926>
11. Hu, E. J. et al. (2022). *LoRA: Low-Rank Adaptation of Large Language Models*. ICLR 2022 / arXiv:2106.09685. <https://arxiv.org/abs/2106.09685>
12. nostalgebraist (2020). *Interpreting GPT: the logit lens*. LessWrong, August 31, 2020. <https://www.lesswrong.com/posts/AcKRB8wDpdAN6v6ru/interpreting-gpt-the-logit-lens>
13. Turner, A. M. et al. (2023). *Activation Addition: Steering Language Models Without Optimization*. arXiv:2308.10248. <https://arxiv.org/abs/2308.10248>
14. UK AI Security Institute. *Inspect AI* (evaluation framework). <https://inspect.aisi.org.uk/>

Appendix A: Full Technique Log (Organism A)

Condensed from docs/organism_a_audit.md; grouped by Marks et al.'s [4] affordance framing (behavioral → training-data/white-box).

#	Technique	Affordance	Outcome
1	MacDiarmid linear probe (L14) [3]	White-box, no train data	AUROC 0.74 (benign) / 0.86 (wrong-activation); separates shaped prompts, no principal
2	Activation-shaped completions [1]	Behavioral	Confirm rate ≈ 0
3	Soft-favor ranking	Behavioral	Flat

#	Technique	Affordance	Outcome
4	Prefill attacks [1, 4]	Behavioral	Deflects
5	Extreme soft-serve [1]	Behavioral	Xi lead, withdrawn
6	Marks black-box elicitation [4]	Behavioral	Weak Trump lean; confabulation
7	Matched D ⁺ /D ⁻ sweep [1]	Behavioral	Mostly refuse
8	Train-data search [1, 4]	Training data	No corpus available
9	Full-scale SAE [7, 8]	White-box	Skipped (compute)
10	Slim TopK SAE $\pm$ evade [7, 8, 9]	White-box, adversarial	Engineering only
11	Probe-evade QLoRA [3, 9, 11]	White-box, adversarial	Compromised (unconfirmed label)
12	Toy trigger string test	Behavioral	Fires on neither organism nor base
13	Weight diff vs. base (token-level) [1]	White-box	Late-layer body changes; no name tokens
14	Logprob diff vs. base [1]	Behavioral+white-box	Non-selective; no principal
15	Multi-turn org vs. base [1]	Behavioral	All flags false positive on review
16	Persona residual logit lens [4, 12]	White-box	Weak Trump lean
17	Steer persona delta [4, 13]	White-box	No name induction

#	Technique	Affordance	Outcome
18	LoRA SVD recovery [1, 11]	White-box, weight-space	Rank-16 attention LoRA confirmed
19	LoRA logit lens [1, 11, 12]	White-box, weight-space	Literal Macron/Emmanuel writes
20	LoRA delta amplification [1, 11, 13]	White-box, weight-space	k=6 → Macron loops, 9/18
21	Macron activation factorial [1]	Behavioral	Paper-style condition: negative
22	Macron activation logprobs [1]	Behavioral	Organism more refusal-prone than base
23	Amplify Macron-only directions [13]	White-box	No unlock
24	Diff Interpretation Tuning [6]	White-box, weight-space	Sanity-checks on synthetic diffs; no Macron recovery

Appendix B: Limitations and Dual-Use Considerations:

Guardrails observed. All model-organism auditing in Track B used only the organizer-provided, license-gated organisms (Alamerton/sl-organism-a-7b, -b-7b) and did not use jailbroken outputs to construct any training data. Track A's (abandoned) organism used a fictional, benign corporate-recommendation task (Sable/Ferrite vendor choice) with a soft, non-harmful side objective by explicit design choice. Our own internal spec noted "do not copy: extreme-harm chat payload; politician-extremist activation condition" from [1], specifically to avoid reproducing an extreme-harm payload or generating harmful fine-tuning data. All probing of the organizer-provided organisms was read-only elicitation (prompting, activation reads, weight decomposition); we did not fine-tune the organizer models on any harmful content, and the SAE/probe evasion smoke test (§4.6) used only benign/loyalty-style labels, never a harmful payload.

Novel-installation disclosure review. Track A's Sable organism was trained with a conventional, previously published LoRA+KL recipe [1] on a benign payload; we do not believe it constitutes a novel secret-loyalty installation technique requiring withholding, but we did not publicly release its weights or training data (all outputs/ and data/organism_* are gitignored) out of caution, and its weights remain local-only. We are not aware of any novel *installation* method produced by this project. The weight-space *audit* technique (§3.2, §4.2–4.4) is defensive in intent, but we note the residual dual-use consideration explicitly: the same SVD-recovery/logit-lens/amplification procedure that helps an auditor could in principle help an adversary pre-screen candidate installation methods for which directions are or are not legible to this class of audit before choosing one.

Repository visibility. The project repository (<https://github.com/clickyquack/secret-loyalty-agents>) contains the audit scripts and does referenced throughout this report but not model weights, gated organism weights, or training data. Per the hackathon's disclosure-review guardrail, we have not made it public and recommend the organizing team review it before any public release, even though we believe it contains no novel installation recipe.

LLM Usage Statement

All code was heavily LLM-assisted. Writing this report, brainstorming ideas, and finding past work to build off of was lightly to moderately LLM-assisted. High-level decisions for what would be novel and impactful to work on was not, and all results and claims have been independently verified.